

IN THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application.

LISTING OF CLAIMS:

1. (currently amended) In a computer network having a plurality of nodes, a file replication system comprising:

means, distributed on each of said nodes, capable of receiving [[a]] an updated file in any one of said nodes; [[and,]]

means, distributed throughout said plurality of nodes and responsive to receiving said file in a certain one of said nodes, for replicating said updated file in all other of said nodes in a manner that is network-topology independent and avoids a single point of failure;

wherein said certain one of said nodes is the originator node and said replicating means further comprises:

means for establishing another of said nodes as master node, said plurality of nodes except for both said originator node and said master node being slave nodes;

means for storing said updated file on said master node as a backup file;
and,

in each of said slave nodes, means for updating a particular file
corresponding to said updated file;

means for associating a file version variable with said updated file;

said storing means comprising:

local workspace means for receiving said updated file in said master

node;

means for error checking said file version variable to confirm validity of

said file version variable;

global workspace means adapted to receive said updated file from said

local workspace means; and,

means, responsive to operation of said error checking means confirming

said validity, for transferring said updated file to said global workspace means;

and,

means, responsive to operation of said transferring means, for communicating

both creation of said backup file to said originator node and availability of said backup file to said slave nodes.

2-5. (canceled)

6. (currently amended) The system of claim [[5]] 1 and further comprising:

means, included within said master node, for communicating both creation of said backup file to said originator node and availability of said backup file to said slave nodes.

7. (original) The system of claim 6 and further comprising:

said originator node including means, responsive to operation of said communicating means, for publishing a representation of said updated file to said slave nodes; and

means, within each of said slave nodes and responsive to operation of said publishing means, for commanding its respective node to obtain said updated file from said originator node.

8. (original) The system of claim 6 and wherein operation of said communicating means communicating said creation of said backup file to said originator node comprises a success note.

9. (original) The system of claim 7 and further comprising:

said originator node including means for establishing a updated file version variable as said representation of said file; and

said publishing means includes means for publishing said updated file version variable to said slave nodes.

10. (original) The system of claim 9 and further comprising:

said replicating means including means for establishing a particular file version variable corresponding to said particular file;

object observer means, in said each of said slave nodes, for observing change from said particular file version variable to said updated file version variable; and,

said commanding means including means, responsive to operation of said object
observer means, for downloading said updated file from said originator node into said
particular file in said each of said slave nodes.

11. (currently amended) The system of claim [[5]] 1 and wherein ~~said replicating~~
~~means comprises:~~

said local workspace means includes other means for receiving said updated file
in said originator node; and,

said global workspace means, ~~operatively coupled to said local workspace means,~~
includes other means for receiving said updated file from said ~~local workspace~~ other
means in preparation to download said updated file to any of said slave nodes upon
request from said any of said slave nodes.

12. (original) The system of claim 11 and wherein said replicating means further
comprises:

at least one data file including a data word with its corresponding file version
variable.

13. (canceled)

14. (original) The system of claim 11 and wherein said local workspace means
further includes:

multiple source means for receiving said updated file from a new-data-supplier group consisting of a network user, a security provider, and other providers.

15. (original) The system of claim 14 and wherein said multiple source receiving means includes additional means for receiving additional updated files from said new-data-supplier group.

16. (currently amended) The system of claim [[2]] 1 and wherein said receiving means is further capable of receiving multiple updated files, each of said files being received from a different network user.

17. (currently amended) In a computer network having a plurality of nodes, a file replication system comprising:

means, distributed on each of said nodes, capable of receiving an updated file in any one of said nodes;

means, distributed throughout said plurality of nodes and responsive to receiving said file in a certain one of said nodes, for replicating said updated file in all other of said nodes in a manner that is network-topology independent and avoids a single point of failure;

wherein said certain one of said nodes is the originator node and said replicating means further comprises:

means for establishing another of said nodes as master node, said plurality of nodes except for both said originator node and said master node being slave nodes;

means for storing said updated file on said master node as a backup file;

in each of said slave nodes, means for updating a particular file corresponding to said updated file; and

~~The system of claim 5 wherein said replicating means includes~~ means for establishing a particular file version variable corresponding to said particular file, ~~said system further comprising;~~

polling means for allowing said each of said slave nodes to periodically poll said master node to determine if said particular file contents matches said updated file contents; and,

synchronizing means, responsive to operation of said polling means determining that said particular file contents do not match said updated file contents, for synchronizing said particular file contents with said updated file contents.

18. (original) The system of claim 17 further comprising:

means for establishing a DPS version number to identify the current version of DPS in said network;

means for establishing a DDB version number to identify the current version of DDB in said network;

first means for comparing said DPS version number on said each of said slave nodes with said DPS version number on said master node to obtain a respective DPS version number match;

said synchronizing means,

responsive to operation of said first comparing means obtaining said respective DPS version number match on each of certain of said slave nodes for terminating further operation of said synchronizing means with respect to the current said poll on said each of said certain of said slave nodes, and

responsive to operation of said first comparing means not obtaining said respective DPS version number match on each of the remainder of said slave nodes for achieving said DDB version number match on said each of the remainder of said slave nodes;

second means, responsive to operation of said first comparing means not obtaining said respective DPS version number match, for comparing said particular file version variable on said each of the remainder of said slave nodes with said updated file version variable on said master node to obtain a respective file version variable match; and,

said synchronizing means,

responsive to operation of said second comparing means obtaining a file version variable match on each of a portion of said remainder of said slave nodes for terminating operation of said synchronizing means with respect to the current said poll on said each of said portion, and

responsive to operation of said second means not obtaining a file version variable match on each of the remaining portion of said remainder of said slave nodes for achieving said file version variable match on said each of the remaining portion;

whereby said particular file contents matches said updated file contents in said each of said slave nodes.

19. (currently amended) In a computer network having a plurality of nodes, a file replication system comprising:

means, distributed on each of said nodes, capable of receiving an updated file in any one of said nodes;

means, distributed throughout said plurality of nodes and responsive to receiving said file in a certain one of said nodes, for replicating said updated file in all other of said nodes in a manner that is network-topology independent and avoids a single point of failure;

wherein said certain one of said nodes is the originator node and said replicating means further comprises:

means for establishing another of said nodes as master node, said plurality of nodes except for both said originator node and said master node being slave nodes;

means for storing said updated file on said master node as a backup file;

in each of said slave nodes, means for updating a particular file corresponding to said updated file; and

~~The system of claim 5 wherein said replicating means includes~~ means for establishing a particular file version variable corresponding to said particular file, and wherein each of said plurality of nodes is a storage system having storage media on

which both said particular file version variable and said particular file contents are stored;
~~said system further comprising;~~

polling means, for comparing said particular file version variable stored on said storage media in said each of said nodes with said particular file version variable stored elsewhere in its respective node to determine a particular file version variable match for said each of said nodes; and

synchronizing means, responsive to operation of said polling means determining said particular file version variable match was not achieved for certain of said nodes for achieving said particular file version variable match for each of said certain nodes.

20. (original) The system of claim 19 and wherein said storage media is at least one storage disk.

21. (original) The system of claim 7 further comprising:

means, responsive to operation of said commanding means not obtaining said updated file from said originator node, for further commanding said respective node to obtain said backup file from said master node.

22. (currently amended) The system of claim [[13]] 1 further comprising:

means, responsive to operation of said error checking means not confirming said validity, for flagging an error, stopping operation of said file replicating means on said updated file, and preparing said file replication system to receive a next successive updated file.

23. (currently amended) A computer program product for use in a computer network having a plurality of nodes, said computer program product including a tangible computer usable medium having computer readable program code thereon for file replication, said program code comprising:

program code, distributed on each of said nodes, capable of receiving [[a]] an updated file in any one of said nodes; [[and,]]

program code, distributed throughout said plurality of nodes and responsive to receiving said file in a certain one of said nodes, for replicating said updated file in all other of said nodes in a manner that is network-topology independent;

wherein said certain one of said nodes is the originator node and said replicating program code further comprises:

program code for establishing another of said nodes as master node, said plurality of nodes except for both said originator node and said master node being slave nodes;

program code for storing said updated file on said master node as a backup file; and,

in each of said slave nodes, program code for updating a particular file corresponding to said updated file;

program code for associating a file version variable with said updated file;

said storing program code comprising:

local workspace program code for receiving said updated file in said
master node;
program code for error checking said file version variable to confirm
validity of said file version variable;
global workspace program code adapted to receive said updated file from
said local workspace program code; and,
program code, responsive to operation of said error checking program
code confirming said validity, for transferring said updated file to said global
workspace program code; and,
program code, responsive to operation of said transferring program code, for
communicating both creation of said backup file to said originator node and availability
of said backup file to said slave nodes.

24. (canceled)

25. (currently amended) The computer program product of claim [[24]] 23 and
wherein said replicating program code includes:

additional replication program code for replicating said updated file in all other of
said nodes in a manner that avoids a single point of failure.

26-27. (canceled)

28. (currently amended) The computer program product of claim ~~[[27]]~~ 23 and further comprising:

program code, included within said master node, for communicating both creation of said backup file to said originator node and availability of said backup file to said slave nodes.

29. (original) The computer program product of claim 28 and further comprising:

said originator node including program code, responsive to operation of said communicating program code, for publishing a representation of said updated file to said slave nodes; and

program code, within each of said slave nodes and responsive to operation of said publishing program code, for commanding its respective node to obtain said updated file from said originator node.

30. (original) The computer program product of claim 28 and wherein operation of said communicating program code communicating said creation of said backup file to said originator node comprises a success note.

31. (original) The computer program product of claim 29 and further comprising:

said originator node including program code for establishing an updated file version variable as said representation of said updated file; and

said publishing program code includes program code for publishing said updated file version variable to said slave nodes.

32. (original) The computer program product of claim 31 and further comprising:
said replicating program code including program code for establishing a particular file version variable corresponding to said particular file;
object observer program code, in said each of said slave nodes, for observing change from said particular file version variable to said updated file version variable; and,
said commanding program code including program code, responsive to operation of said object observer program code, for downloading said updated file from said originator node into said particular file in said each of said slave nodes.
33. (currently amended) The computer program product of claim ~~[[27]]~~ 23 and wherein said replicating program code comprises:
local workspace program code for receiving said updated file in said originator node; and,
global workspace program code, operatively coupled to said local workspace program code, for receiving said updated file from said local workspace program code in preparation to download said updated file to any of said slave nodes upon request from said any of said slave nodes.
34. (original) The computer program product of claim 33 and wherein said replicating program code further comprises:
at least one data file including a data word with its corresponding file version variable.

35. (canceled)

36. (original) The computer program product of claim 33 and wherein said local workspace program code further includes:

multiple source program code for receiving said updated file from a new-data-supplier group consisting of a network user, a security provider, and other providers.

37. (original) The computer program product of claim 36 and wherein said multiple source receiving program code includes additional program code for receiving additional updated files from said new-data-supplier group.

38. (currently amended) The computer program product of claim ~~[[24]]~~ 23 and wherein said receiving program code is further capable of receiving multiple updated files, each of said files being received from a different network user.

39. (currently amended) A computer program product for use in a computer network having a plurality of nodes, said computer program product including a tangible computer usable medium having computer readable program code thereon for file replication, said program code comprising:

program code, distributed on each of said nodes, capable of receiving an updated file in any one of said nodes;

program code, distributed throughout said plurality of nodes and responsive to receiving said file in a certain one of said nodes, for replicating said updated file in all other of said nodes in a manner that is network-topology independent;

wherein said certain one of said nodes is the originator node and said replicating program code further comprises:

program code for establishing another of said nodes as master node, said plurality of nodes except for both said originator node and said master node being slave nodes;

program code for storing said updated file on said master node as a backup file; and,

in each of said slave nodes, program code for updating a particular file corresponding to said updated file; and

~~The computer program product of claim 27 wherein said replicating program code includes~~ program code for establishing a particular file version variable corresponding to said particular file, ~~said computer program product further comprising;~~

polling program code for allowing said each of said slave nodes to periodically poll said master node to determine if said particular file contents matches said updated file contents; and,

synchronizing program code, responsive to operation of said polling program code determining that said particular file contents do not match said updated file contents, for synchronizing said particular file contents with said updated file contents.

40. (original) The computer program product of claim 39 further comprising:

program code for establishing a DPS version number to identify the current version of DPS in said network;

program code for establishing a DDB version number to identify the current version of DDB in said network;

first program code for comparing said DPS version number on said each of said slave nodes with said DPS version number on said master node to obtain a respective DPS version number match;

said synchronizing program code,

responsive to operation of said first comparing program code obtaining said respective DPS version number match on each of certain of said slave nodes for terminating further operation of said synchronizing program code with respect to the current said poll on said each of said certain of said slave nodes, and

responsive to operation of said first comparing program code not obtaining said respective DPS version number match on each of the remainder of said slave nodes for achieving said DDB version number match on said each of the remainder of said slave nodes;

second program code, responsive to operation of said first comparing program code not obtaining said respective DPS version number match, for comparing said particular file version variable on said each of the remainder of said slave nodes with said updated file version variable on said master node to obtain a respective file version variable match; and,

said synchronizing program code,

responsive to operation of said second comparing program code obtaining a file version variable match on each of a portion of said remainder of said slave nodes for terminating operation of said synchronizing program code with respect to the current said poll on said each of said portion, and

responsive to operation of said second program code not obtaining a file version variable match on each of the remaining portion of said remainder of said slave nodes for achieving said file version variable match on said each of the remaining portion;

whereby said particular file contents matches said updated file contents in said each of said slave nodes.

41. (currently amended) A computer program product for use in a computer network having a plurality of nodes, said computer program product including a tangible computer usable medium having computer readable program code thereon for file replication, said program code comprising:

program code, distributed on each of said nodes, capable of receiving an updated file in any one of said nodes;

program code, distributed throughout said plurality of nodes and responsive to receiving said file in a certain one of said nodes, for replicating said updated file in all other of said nodes in a manner that is network-topology independent;

wherein said certain one of said nodes is the originator node and said replicating program code further comprises:

program code for establishing another of said nodes as master node, said plurality of nodes except for both said originator node and said master node being slave nodes;

program code for storing said updated file on said master node as a backup file; and,

in each of said slave nodes, program code for updating a particular file corresponding to said updated file; and

~~The computer program product of claim 27 wherein said replicating program code includes~~ program code for establishing a particular file version variable corresponding to said particular file, and wherein each of said plurality of nodes is a storage computer program product having storage media on which both said particular file version variable and said particular file contents are stored, ~~said computer program product further comprising;~~

polling program code, for comparing said particular file version variable stored on said storage media in said each of said nodes with said particular file version variable stored elsewhere in its respective node to determine a particular file version variable match for said each of said nodes; and

synchronizing program code, responsive to operation of said polling program code determining said particular file version variable match was not achieved for certain of said nodes for achieving said particular file version variable match for each of said certain nodes.

42. (original) The computer program product of claim 41 and wherein said storage media is at least one storage disk.

43. (original) The computer program product of claim 29 further comprising:
program code, responsive to operation of said commanding program code not obtaining said updated file from said originator node, for further commanding said respective node to obtain said backup file from said master node.

44. (currently amended) The computer program product of claim ~~[[35]]~~ 23 further comprising:

program code, responsive to operation of said error checking program code not confirming said validity, for flagging an error, stopping operation of said file replicating program code on said updated file, and preparing said file replication computer program product to receive a next successive updated file.

45. (currently amended) In a computer network having a plurality of nodes, a file replication method executed by a computer, said method comprising:

arranging for receipt of ~~[[a]]~~ an updated file in any one of said nodes;

receiving said updated file in a certain one of said nodes; ~~[[and,]]~~

replicating said file in all other of said nodes in a manner that is network-topology independent;

wherein said certain one of said nodes is the originator node and said replicating further comprises:

establishing another of said nodes as master node, said plurality of nodes except for both said originator node and said master node being slave nodes;

storing said updated file on said master node as a backup file; and,

in each of said slave nodes, updating a particular file corresponding to said updated file;

associating a file version variable with said updated file;

said storing comprising:

receiving said updated file in local workspace of said master node;

error checking said file version variable to confirm validity of said file version variable;

adapting global workspace to receive said updated file from said local workspace; and,

transferring said updated file to said global workspace responsive to said error checking confirming said validity; and,

communicating both creation of said backup file to said originator node and availability of said backup file to said slave nodes responsive to said transferring.

46. (canceled)

47. (currently amended) The method of claim [[46]] 45 and wherein said replicating program code includes:

replicating said updated file in all other of said nodes in a manner that avoids a single point of failure.

48-49. (canceled)

50. (currently amended) The method of claim [[49]] 45 further comprising:
communicating both creation of said backup file to said originator node and availability of said backup file to said slave nodes.

51. (original) The method of claim 50 further comprising:
publishing a representation of said updated file to said slave nodes; and
within each of said slave nodes, commanding its respective node to obtain said updated file from said originator node.

52. (original) The method of claim 50 and wherein communicating said creation of said backup file to said originator node comprises a success note.

53. (original) The method of claim 51 further comprising:
establishing an updated file version variable as said representation of said updated file; and
publishing said updated file version variable to said slave nodes.

54. (original) The method of claim 53 further comprising:

establishing a particular file version variable corresponding to said particular file;
observing change from said particular file version variable to said updated file
version variable; and,
downloading said updated file from said originator node into said particular file in
said each of said slave nodes.

55. (currently amended) The method of claim [[49]] 45 further comprising:
receiving said updated file in local workspace of said originator node; and,
receiving said updated file into global workspace from said local workspace in
preparation to download said updated file to any of said slave nodes upon request from
said any of said slave nodes.

56. (original) The method of claim 55 and wherein said replicating further comprises:
including a data word with its corresponding file version variable in at least one
data file.

57. (canceled)

58. (original) The method of claim 55 further comprising:
receiving said updated file from a new-data-supplier group consisting of a
network user, a security provider, and other providers.

59. (original) The method of claim 58 further comprising:

receiving additional updated files from said new-data-supplier group.

60. (currently amended) The method of claim ~~[[46]]~~ 45 further comprising:

receiving multiple updated files, each of said files being received from a different network user.

61. (currently amended) In a computer network having a plurality of nodes, a file replication method executed by a computer, said method comprising:

arranging for receipt of an updated file in any one of said nodes;

receiving said updated file in a certain one of said nodes;

replicating said file in all other of said nodes in a manner that is network-topology independent;

wherein said certain one of said nodes is the originator node and said replicating further comprises:

establishing another of said nodes as master node, said plurality of nodes except for both said originator node and said master node being slave nodes;

storing said updated file on said master node as a backup file; and,

in each of said slave nodes, updating a particular file corresponding to said updated file;

~~The method of claim 49 further comprising:~~

establishing a particular file version variable corresponding to said particular file;

said each of said slave nodes periodically polling said master node to determine if said particular file contents matches said updated file contents;

determining that said particular file contents do not match said updated file contents; and,

synchronizing said particular file contents with said updated file contents.

62. (original) The method of claim 61 further comprising:

establishing a DPS version number to identify the current version of DPS in said network;

establishing a DDB version number to identify the current version of DDB in said network;

first comparing said DPS version number on said each of said slave nodes with said DPS version number on said master node to obtain a respective DPS version number match;

said synchronizing,

responsive to said first comparing obtaining said respective DPS version number match on each of certain of said slave nodes, terminating further operation of said synchronizing with respect to the current poll on said each of said certain of said slave nodes, and

further responsive to said first comparing not obtaining said respective DPS version number match on each of the remainder of said slave nodes, achieving said DDB version number match on said each of the remainder of said slave nodes;

second comparing said particular file version variable on said each of the remainder of said slave nodes with said updated file version variable on said master node to obtain a respective file version variable match responsive to said first comparing not obtaining said respective DPS version number match; and,

said synchronizing,

further responsive to said second comparing obtaining a file version variable match on each of a portion of said remainder of said slave nodes, terminating operation of said synchronizing with respect to said current poll on said each of said portion, and

further responsive to said second comparing not obtaining a file version variable match on each of the remaining portion of said remainder of said slave nodes, achieving said file version variable match on said each of the remaining portion;

whereby said particular file contents matches said updated file contents in said each of said slave nodes.

63. (currently amended) In a computer network having a plurality of nodes, a file replication method executed by a computer, said method comprising:

arranging for receipt of an updated file in any one of said nodes;

receiving said updated file in a certain one of said nodes;

replicating said file in all other of said nodes in a manner that is network-topology independent;

wherein said certain one of said nodes is the originator node and said replicating further comprises:
establishing another of said nodes as master node, said plurality of nodes except for both said originator node and said master node being slave nodes;
storing said updated file on said master node as a backup file; and,
in each of said slave nodes, updating a particular file corresponding to said updated file; and

~~The method of claim 49~~ wherein each of said plurality of nodes is a storage system having storage media and wherein said replicating further comprises:
establishing a particular file version variable corresponding to said particular file;
storing said particular file version variable and said particular file contents on said storage media to obtain stored particular file version variable and contents;
[[third]] comparing said particular file version variable stored on said storage media in said each of said nodes with said particular file version variable stored elsewhere in its respective node to determine a particular file version variable match for said each of said nodes; and,
synchronizing, responsive to[[third]] comparing determining said particular file version variable match was not achieved for certain of said nodes, achieving said particular file version variable match for each of said certain nodes.

64. (original) The method of claim 63 and wherein said storage media is at least one storage disk.

65. (original) The method of claim 51 further comprising:

further commanding said respective node to obtain said backup file from said master node responsive to said commanding not obtaining said updated file from said originator node.

66. (currently amended) The method of claim [[57]] 45 responsive to said error checking not confirming said validity, said method further comprising:

flagging an error;

stopping said file replicating on said updated file; and,

preparing said file replication method to receive a next successive updated file.

67. (canceled)

68. (original) A method for synchronizing data in a network having a master node and at least one slave node comprising:

establishing a DPS in said master node and said at least one slave node;

for each said slave node, periodically polling said master node to determine if the version number of said DPS in said master node matches the version number of said DPS in said at least one slave node;

for each said slave node, terminating said method if said master node DPS version number matches said at least one slave node DPS version number; and,

if said master node DPS version number does not match said at least one slave node DPS version number thereby providing a mismatch:

first determining if said mismatch is due to a DDB version number mismatch only and, if so, first replicating IP addresses of said at least one slave node and terminating said method; and,

if said mismatch is not due to a DDB version number mismatch only, second determining if said mismatch is due to a FVV mismatch only and, if so, second replicating new files associated with a new said FVV and terminating said method; and,

if said mismatch is due to both said DDB version number mismatch and said FVV mismatch, performing said first replicating and said replicating and terminating said method.

69. (original) A method for synchronizing data on storage media in a network having a master node and at least one other node including slave node and originator node, said method comprising:

establishing a DPS in said master node and said at least one other node;

for each node in said network, periodically comparing each FVV stored in its respective said storage media with its corresponding FVV stored in its DPS;

if there is a match between said storage media FVV and its corresponding said DPS FVV, terminating said method;

if there is no match between said storage media FVV and its corresponding said DPS FVV thereby providing a first mismatch, first determining if said first mismatch occurred on said master node or on said at least one other node;

if said first mismatch occurred on said at least one other node second determining if said first mismatch is due to a missing file or an extra file on said other node storage media; and

if due to said missing file, retrieving said missing file from said originator node or said master node and terminating said method; and,

if due to said extra file, ignoring said extra file and deleting said extra file from said media and terminating said method; and,

if said first mismatch occurred on said master node, third determining if said mismatch is due to a missing file or an extra file on said master node storage media; and

if due to said missing file, removing the corresponding file from said DPS in said master node and terminating said method; and,

if due to said extra file, adding said extra file to said DPS in said master node and terminating said method.

70-72. (canceled)